

## Agrarwissenschaften und Ernte

Als Agrarwissenschaftler:in möchten Sie ein Modell entwickeln, das den Ernteertrag eines bestimmten Feldes basierend auf ausgebrachtem Dünger (in Kilogramm pro Hektar) und der durchschnittlichen Niederschlagsmenge während der Wachstumsperiode (in Millimetern) vorhersagen kann. Ihre Forschung konzentriert sich auf die Optimierung landwirtschaftlicher Erträge durch Vorhersagemodelle.

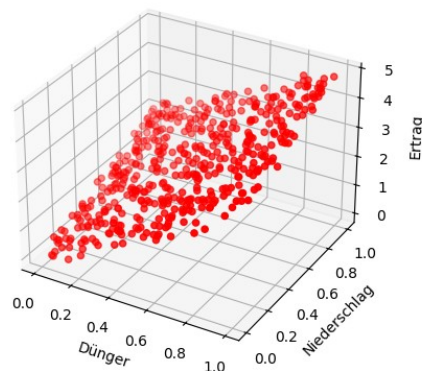
### Aufgabenstellungen:

- Lade die Datei harvest.csv.
- Gib die ersten 5 Zeilen des Datensets auf der Kommandozeile aus:

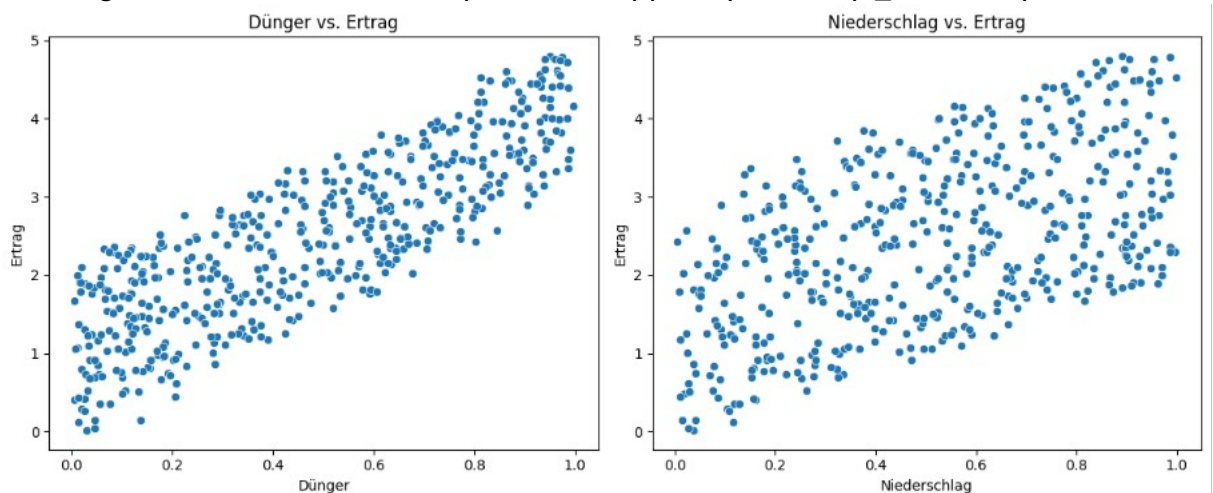
	Dünger	Niederschlag	Ertrag
0	0.374540	0.950714	3.042819
1	0.731994	0.598658	3.259764
2	0.156019	0.155995	0.818065
3	0.058084	0.866176	1.967662
4	0.601115	0.708073	3.275469

- Schreibe ein Programm p1\_regression in Python, welches die Daten wie folgt visualisiert. Nutze die Bibliotheken pandas, numpy, matplotlib, mpl\_toolkits.mplot3d:

3D-Plot der Ertragsdaten



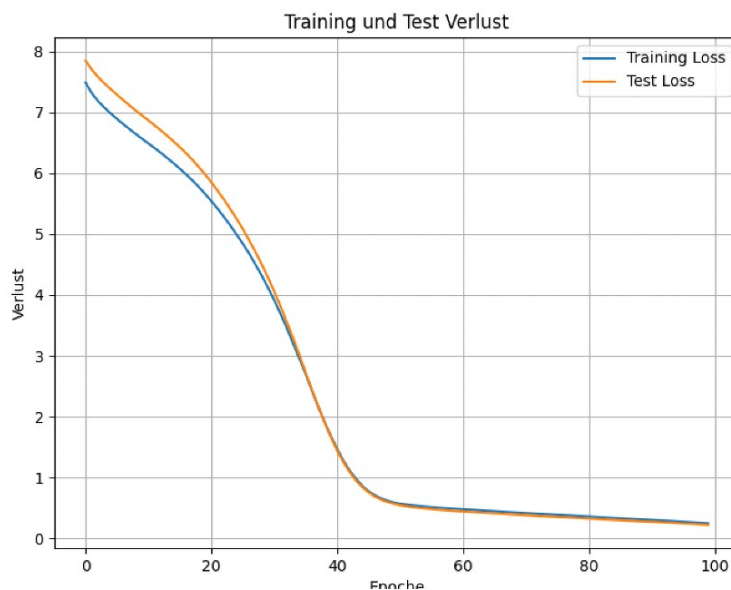
- Nutze seaborn, um in p1\_regression das paarweise Verhältnis folgender Variablen anzuzeigen. Nutze die Bibliotheken pandas, numpy, matplotlib, mpl\_toolkits.mplot3d:



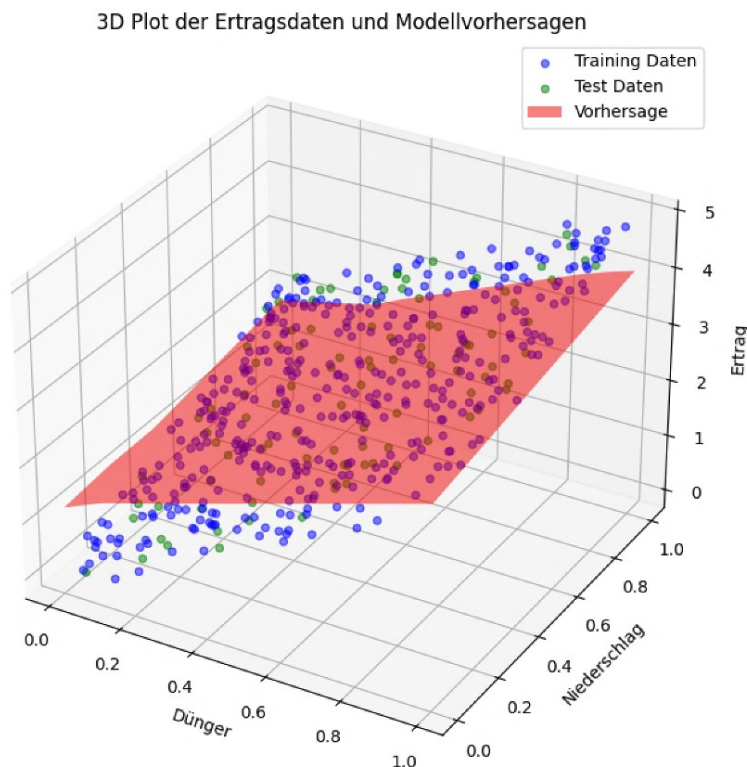
- Randomisiere das Datenset. Teile die Daten in p2\_regression in Trainings- und Testdaten (80% Training, 20% Test). Erstelle in Keras ein Modell mit 2 Hidden Layers mit

jeweils 4 Knoten und Relu-Aktivierung. Nimm den Mean Squared Error als Loss-Funktion, und den Adam-Optimizer. Trainiere das Modell 100 Epochen lang, mit einer Batch-Größe von 50.

- Plote die Loss-Funktionen für Training und Test:



- Plote das Modell und die Daten (getrennt nach Test- und Trainingsdaten):



- Berechne zuletzt die Vorhersagewerte des Modells für folgende Wertepaare: Niederschlag/Dünger: [0.25, 0.25], [0.85, 0.75].

## Dünnschnabelmöwen und Dickhornschafe

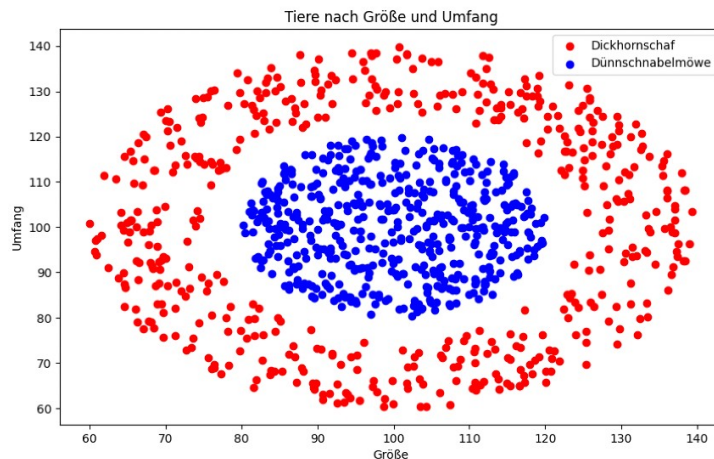
Die Unterscheidung zwischen Dünnschnabelmöwen und Dickhornschafen stellt eine interessante und relevante Aufgabenstellung in der biologischen Forschung und Naturschutz dar. Diese beiden Arten sind nicht nur ökologisch bedeutend, sondern auch faszinierend in ihrem Verhalten und ihren Lebensräumen. Die Fähigkeit, diese beiden Arten zu unterscheiden, ist deshalb essentiell für den weiteren beruflichen Werdegang:

### Aufgabenstellungen:

- Lade die Datei animals.csv.
- Gib die ersten 5 Zeilen des Datensets auf der Kommandozeile aus:

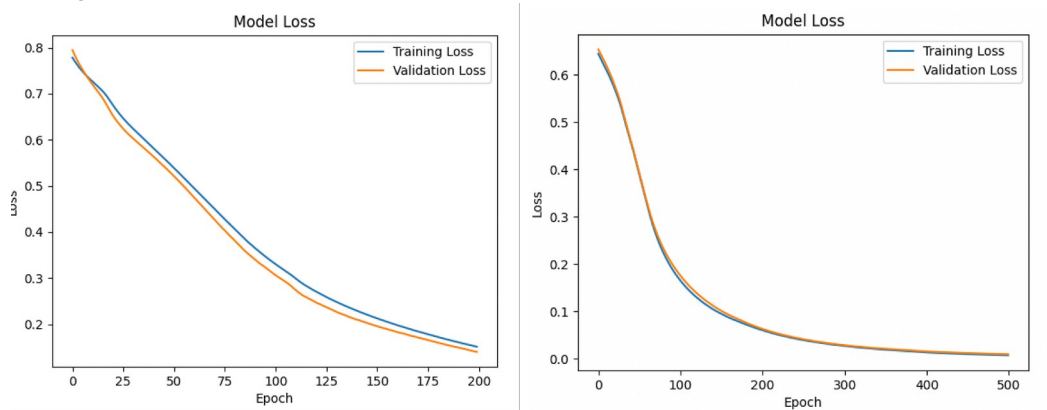
	Groesse	Umfang	Label
0	94.981605	118.028572	Duennschnabelmoewe
1	109.279758	103.946339	Duennschnabelmoewe
2	86.240746	86.239781	Duennschnabelmoewe
3	104.044600	108.322903	Duennschnabelmoewe
4	113.297706	88.493564	Duennschnabelmoewe

- Schreibe ein Programm p1\_classification in Python, welches die Daten wie folgt visualisiert. Nutze die Bibliotheken pandas, matplotlib:

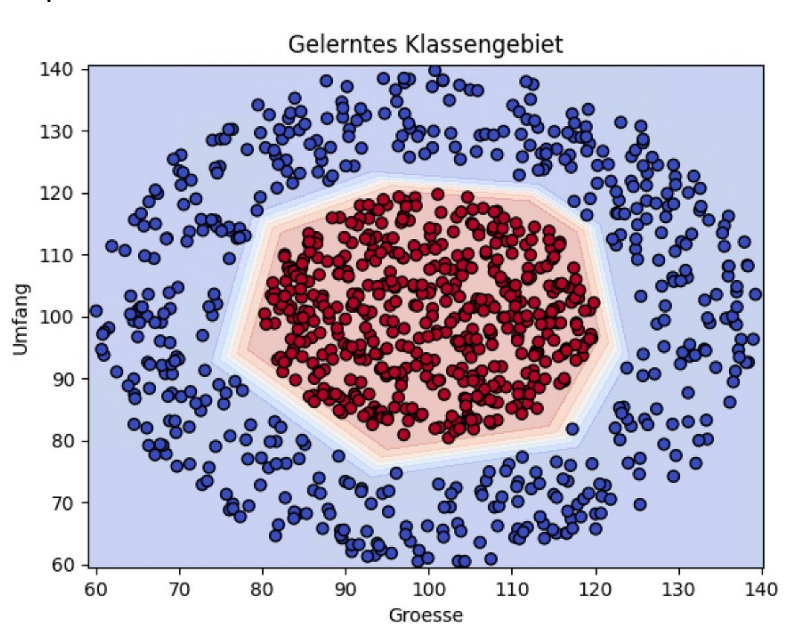


- Randomisiere das Datenset. Kodiere die Labels als numerische Werte, damit sie von Tensorflow verarbeitet werden können. Teile die Daten in p2\_classification in Trainings- und Testdaten (80% Training, 20% Test). Erstelle in Keras ein Modell mit 2 Hidden Layers mit jeweils 4 Knoten und Relu-Aktivierung. Der Output-Knoten soll die Sigmoid-Funktion als Aktivierung haben. Nimm Binary Cross Entropy als Loss-Funktion, und den Adam-Optimizer. Trainiere das Modell 50 (eventuell mehr) Epochen lang, mit einer Batch-Größe von 100.

- Plote die Loss-Funktion über Training und Test. Wie unterscheidet sich der Verlauf für wenige und viele Epochen?



- Plote das gelernte Modell und die Test-Daten: Nutze `contourf` und `scatter` der `matplotlib`:



- Berechne zuletzt die Vorhersagewerte des Modells für folgende Wertepaare:  
Umfang/Groesse: [90, 90], [70, 70].